

Implied Instruction Set  
Computing / Multiple Recurring  
Instruction Set Computing  
The Epoch of New Processor

# Presented By

Suminda Sirinath Salpitikoralala Dharmasena

(E-mail: [sirinath1978m@gmail.com](mailto:sirinath1978m@gmail.com);

Mobile: +94-(0)714 322975;

Telephone: +94-(0)11-5 364614;

Web: [http://sirio.sourceforge.net/.](http://sirio.sourceforge.net/))

# Drawbacks of the Current State of Art

- Von Neumann bottleneck
- Lack of finer level of parallelism
- Multiple cycles: fetch, decode & execute
- Hazards
  - Control
  - Data
- Compiler based hazard management

- Level of compiler control of the processor
- Complexity
- Efficiency
  - Parallel use of FUs
  - CPU Overheads

# Solution

- Reduce the need of hazard management on the CPU
- Eliminate or deduce complexities
  - Pipelining
  - Out of order execution
  - Branch prediction
  - Hazard management
  - Instruction Set

- Eliminate or deduce CPU overheads
  - Micro code
  - Pipeline management
  - Hazard management
  - Cycle reduction (fetch, executed)
- Increase efficiency
  - Use of FUs
    - Ability to use more FUs Concurrently
  - Parallelization of instructions
  - Throughput per unit of Power
  - Throughput per clock cycle

- Compiler Control
  - Instruction re scheduling
  - Cache management
  - Hazard management
  - Branch scheduling

# Applications

- Super computing
- Low power computing
  - Reduced clock cycle
- Application Specific Computing
- AI
- Execution of any inherently parallel algorithms



# Concluding Remarks

- High speed → Throughput
- High efficiency
  - Throughput per unit of Power
  - CPU resource utilization
- Less Complexities
  - Elimination of the need to have complex schemes currently used

Thank You!

Any questions?